

IN THE CLAIMS

1. (Previously Presented) A method for dynamic distributed data caching comprising:

providing a cache community on a first side of a point of presence, the cache community comprising at least one peer, the cache community being associated with each peer having an associated first content portion indicating content obtained from a second side of the point of presence, the content being ~~to be~~ cached by the ~~respective~~ at least one peer;

allowing a client to join the cache community;

updating a peer list associated with the cache community to include the client, the peer list indicating the peers in the cache community;

~~associating a second content portion corresponding to the client with each peer~~ the content with the client based on joinder of the client, ~~the second content portion being distinct from the first content portion;~~

re-allocating the cache storage of the content ~~allocating the first content portion and the second content portion~~ among the peers in the cache community in response to allowing the client to join the community.

2. (Original) The method for dynamic distributed data caching according to Claim 1 and further comprising:

receiving a join request from the client; and

determining whether to allow the client to join the cache community.

3. (Original) The method dynamic distributed data caching according to Claim 2, wherein the join request comprises a CRMSG\_REQUESTTOJOIN data message.

4. (Original) The method for dynamic distributed data caching according to Claim 1, wherein allowing the client to join the cache community comprises:

- generating an allow message;
- associating the peer list with the allow message; and
- communicating the allow message to the client.

5. (Original) The method for dynamic distributed data caching according to Claim 4, wherein allowing the client to joint the cache community comprises:

- generating an allow message comprising the peer list updated to include the client;
- communicating the allow message to the client; and
- communicating the allow message to at least one member associated with the cache community.

6. (Original) The method for dynamic distributed data caching according to Claim 4, wherein the allow message comprises a CRMSG\_UPDATEPEERLIST data message.

7. (Original) The method for dynamic distributed caching according to Claim 4, wherein the peer list associated with the allow message comprises the updated peer list which includes the client.

8. (Previously Presented) The method for dynamic distributed data caching according to Claim 1, wherein the point of presence is an ISP.

9. (Original) The method for dynamic distributed data caching according to Claim 1, wherein a one of the peers comprises a member.

10. (Original) The method for dynamic distributed data caching according to Claim 1, wherein a one of the peers comprises a master.

11. (Currently Amended) The method for dynamic distributed data caching according to Claim 1, wherein associating a ~~respective second content portion~~ the content with the client comprises:

allocating ~~respective second~~ the content portions to peers in the peer list; and

updating an allocation table to indicate the ~~second content portion~~ associated with the peers.

12. (Canceled).

13. (Currently Amended) The method for dynamic distributed data caching according to Claim 11, wherein ~~at least two of the second content portions overlap~~ the content is cached at more than one peer.

14. (Currently Amended) The method for dynamic distributed data caching according to Claim 11, wherein the ~~first and second content portions respectively comprise~~ includes a plurality of internet protocol domain names.

15. (Currently Amended) The method for dynamic distributed data caching according to Claim 1 and further comprising removing the association between the ~~first~~ content portions and the peers.

16. (Currently Amended) A system for dynamic distributed data caching comprising:

logic encoded on storage and operable to:

provide a cache community on a first side of a point of presence, the cache community comprising at least one peer, the cache community being associated with each peer having an associated first content portion indicating content obtained from a second side of the point of presence, the content being ~~to be~~ cached by the ~~respective~~ at least one peer;

allow a client to join the cache community;

update a peer list associated with the cache community to include the client, the peer list indicating the peers in the cache community;

associate the content with a second content portion ~~corresponding to~~ the client ~~with each peer~~ based on joinder of the client, ~~the second content portion being distinct from the first content portion;~~

re-allocate the cache storage of the content ~~allocate the first content portion and the second content portion~~ among the peers in the cache community in response to allowing the client to join the community.

17. (Original) The system for dynamic distributed data caching according to Claim 16, wherein the logic is further operable to:

receive a join request from the client; and

determine whether to allow the client to join the cache community.

18. (Original) The system for dynamic distributed data caching according to Claim 17, wherein the join request comprises a CRMSG\_REQUESTTOJOIN data message.

19. (Original) The system for dynamic distributed data caching according to Claim 16, wherein the logic is further operable to:

- generate an allow message;
- associate the peer list with the allow message;
- communicate the allow message to the client.

20. (Original) The system for dynamic distributed data caching according to Claim 19, wherein the logic is further operable to:

- generate an allow message comprising the peer list updated to include the client;
- communicate the allow message to the client; and
- communicate the allow message to at least one member associated with the cache community.

21. (Original) The system for dynamic distributed data caching according to Claim 19, wherein the allow message comprises a CRMSG\_UPDATEPEERLIST data message.

22. (Original) The system for dynamic distributed caching according to Claim 19, wherein the peer list associated with the allow message comprises the updated peer list which includes the client.

23. (Previously Presented) The system for dynamic distributed data caching according to Claim 16, wherein the point of presence is an ISP.

24. (Original) The system for dynamic distributed data caching according to Claim 16, wherein a one of the peers comprises a member.

25. (Original) The system for dynamic distributed data caching according to Claim 16, wherein a one of the peers comprises a master.

26. (Currently Amended) The system for dynamic distributed data caching according to Claim 16, wherein the logic is further operable to:

allocate ~~respective second content portions~~ the content to peers in the peer list; and

update an allocation table to indicate the ~~second content portion~~ associated with the peers.

27. (Canceled).

28. (Currently Amended) The system for dynamic distributed data caching according to Claim 26, wherein ~~at least two of the second content portions overlap~~ the content is cached at more than one peer.

29. (Currently Amended) The system for dynamic distributed data caching according to Claim 26, wherein the ~~first and second content portions respectively comprise~~ includes a plurality of internet protocol domain names.

30. (Currently Amended) The system for dynamic distributed data caching according to Claim 16, wherein the logic is further operable to remove the association between the ~~first content portions~~ and the peers.

31. (Canceled)

32. (Canceled)

33. (Canceled)

34. (Canceled)

35. (Canceled)

36. (Canceled)

37. (Canceled)

38. (Canceled)

39. (Canceled)

40. (Canceled)

41. (Canceled)

42. (Canceled)

43. (Canceled)

44. (Canceled)

45. (Canceled)

46. (Canceled)

47. (Canceled)

48. (Canceled)

49. (Canceled)

50. (Canceled)

51. (Canceled)

52. (Canceled)

53. (Canceled)

54. (Canceled)

55. (Canceled)

56. (Canceled)

57. (Canceled)

58. (Canceled)

59. (Canceled)

60. (Canceled)

61. (Canceled)

62. (Canceled)

63. (Canceled)

64. (Canceled)

65. (Canceled)



66. (Canceled)

67. (Canceled)

68. (Canceled)

69. (Canceled)

70. (Canceled)

71. (Canceled)

72. (Canceled)

73. (Canceled)

74. (Canceled)

75. (Canceled)

76. (Canceled)

77. (Canceled)

78. (Canceled)

79. (Canceled)

80. (Canceled)

81. (Canceled)

82. (Canceled)

83. (Canceled)

84. (Canceled)

85. (Canceled)

86. (Canceled)

87. (Canceled)

88. (Canceled)

89. (Canceled)

90. (Canceled)

91. (Canceled)

92. (Canceled)

93. (Canceled)

94. (Canceled)

95. (Currently Amended) A method for dynamic distributed data caching comprising:

communicating a community request to an administration module;

receiving a community list from the administration module in response to the community request, the community list including a list of communities;

selecting one of the communities to attempt to join;

generating a join request to attempt to join the selected one of the communities;

receiving an allow message associated with the selected one of the communities;

receiving a peer list associated with the selected one of the communities;

receiving content allocated ~~from peers~~ for storage in caches of peers in the peer list for cache storage re-allocation in response to joining the selected one of the communities; and

providing content for cache storage re-allocation ~~allocation~~ to peers in the peer list in response to joining the selected one of the communities.

96. (Original) The method for dynamic distributed data caching according to Claim 95, wherein the community request comprises a CRMSG\_WAKEUP data message.

97. (Original) The method for dynamic distributed data caching according to Claim 95, wherein the join request comprises a CRMSG\_REQUESTTOJOIN data message.

98. (Currently Amended) A system for dynamic distributed data caching comprising:

logic encoded on storage and operable to:

communicate a community request to an administration module;

receive a community list from the administration module in response to the community request, the community list including a list of communities;

select one of the communities to attempt to join;

generate a join request to attempt to join the selected one of the communities;

receive an allow message associated with the selected one of the communities;

receive a peer list associated with the selected one of the communities;

receive content allocated ~~from peers~~ for storage in caches of peers in the peer list for cache storage re-allocation in response to joining the selected one of the communities; and

provide content for cache storage re-allocation ~~allocation~~ to peers in the peer list in response to joining the selected one of the communities.

99. (Original) The system for dynamic distributed data caching according to Claim 98, wherein the community request comprises a CRMSG\_WAKEUP data message.

100. (Original) The system for dynamic distributed data caching according to Claim 98, wherein the join request comprises a CRMSG\_REQUESTTOJOIN data message.

101. (Previously presented) A system for dynamic distributed data caching comprising:

means for providing a cache community on a first side of a point of presence, the cache community comprising at least one peer, the cache community being associated with each peer ~~having an associated first content portion indicating content obtained from a second side of the point of presence, the content being~~ to be cached by the respective at least one peer;

means for allowing a client to join the cache community;

means for updating a peer list associated with the cache community to include the client, the peer list indicating the peers in the cache community;

means for associating ~~a second content portion~~ corresponding to the content with the client ~~with each peer~~ based on joinder of the client, ~~the second content portion being distinct from the first content portion;~~

means for re-allocating the cache storage of the content ~~allocating the first content portion and the second content portion~~ among the peers in the cache community in response to allowing the client to join the community.

102. (Canceled)

103. (Canceled)

104. (Canceled)

105. (Previously Presented) A system for dynamic distributed data caching comprising:

means for communicating a community request to an administration module;

means for receiving a community list from the administration module in response to the community request, the community list including a list of communities;

means for selecting one of the communities to attempt to join;

means for generating a join request to attempt to join the selected one of the communities;

means for receiving an allow message associated with the selected one of the communities;

means for receiving a peer list associated with the selected one of the communities;

means for receiving content allocated ~~from peers~~ for storage in caches of peers in the peer list for cache storage re-allocation in response to joining the selected one of the communities; and

means for providing content for cache storage re-allocation ~~allocation~~ to peers in the peer list in response to joining the selected one of the communities.

Please cancel Claims 12 and 27 as indicated above without prejudice or disclaimer.